Info 206

Lab Introduction

http://goo.gl/forms/UeUjqFq22W

Agenda

- General lab overview & expectations
- General tips and best practices
- Getting Started on homework #1

What to expect during your lab section

- The main objective for every lab is to help you get started on the homework
- We do not plan to cover new material unless it is relevant to your successful completion of the homework
- If we finish early, additional time will be spent answering individual questions

What lab section will not cover

- Set up and use of Integrated Development Environment (IDE)
- Debugging tools - IDE debuggers, GDB, pdb, etc.
- Unit testing

Attendance & Participation

- If you do not want to be here, dont come
- You will not be penalized attendance/participation
- Just let us know

Grading

- Grading rubric for each homework assignment will be discussed in lab and posted to Piazza

Homework Submission

- We would love to use GitHub.... but, FERPA
- https://www.ischool.berkeley.edu/uploader
- Late assignments are not accepted

Plagiarism

- We are using plagiarism detection software
- MOSS (Measure of Software Similarity)
- Changing variable and method names, re-arranging code, and other common techniques will not fool MOSS
- Just don't do it!

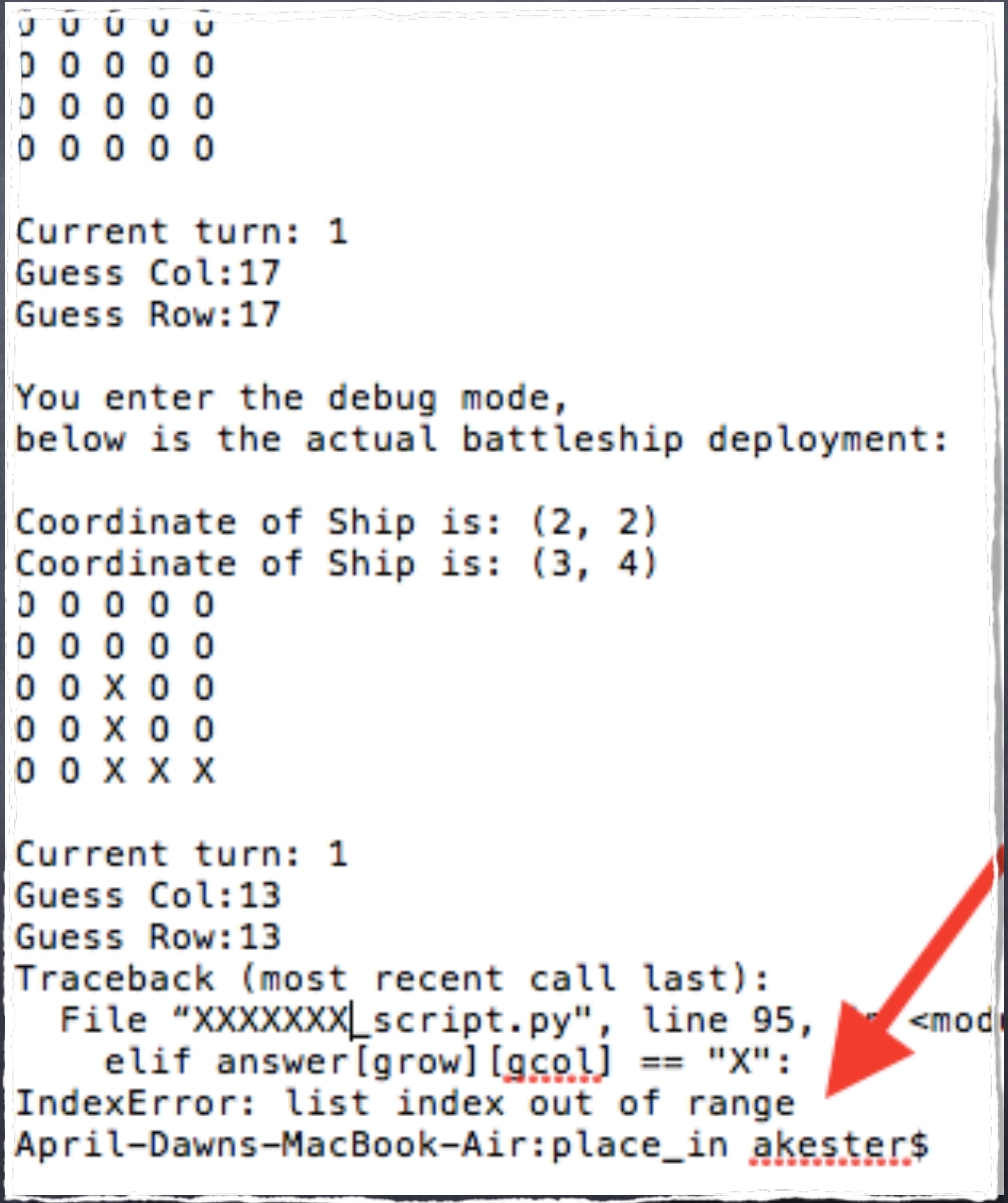
Feedback regarding "place-in" exam

- Follow directions - implement all functionality
- Do not change output strings
- Do not change the parameters or make the program more complex
- Test, test, test, test, test!!!!

Your program should exit gracefully

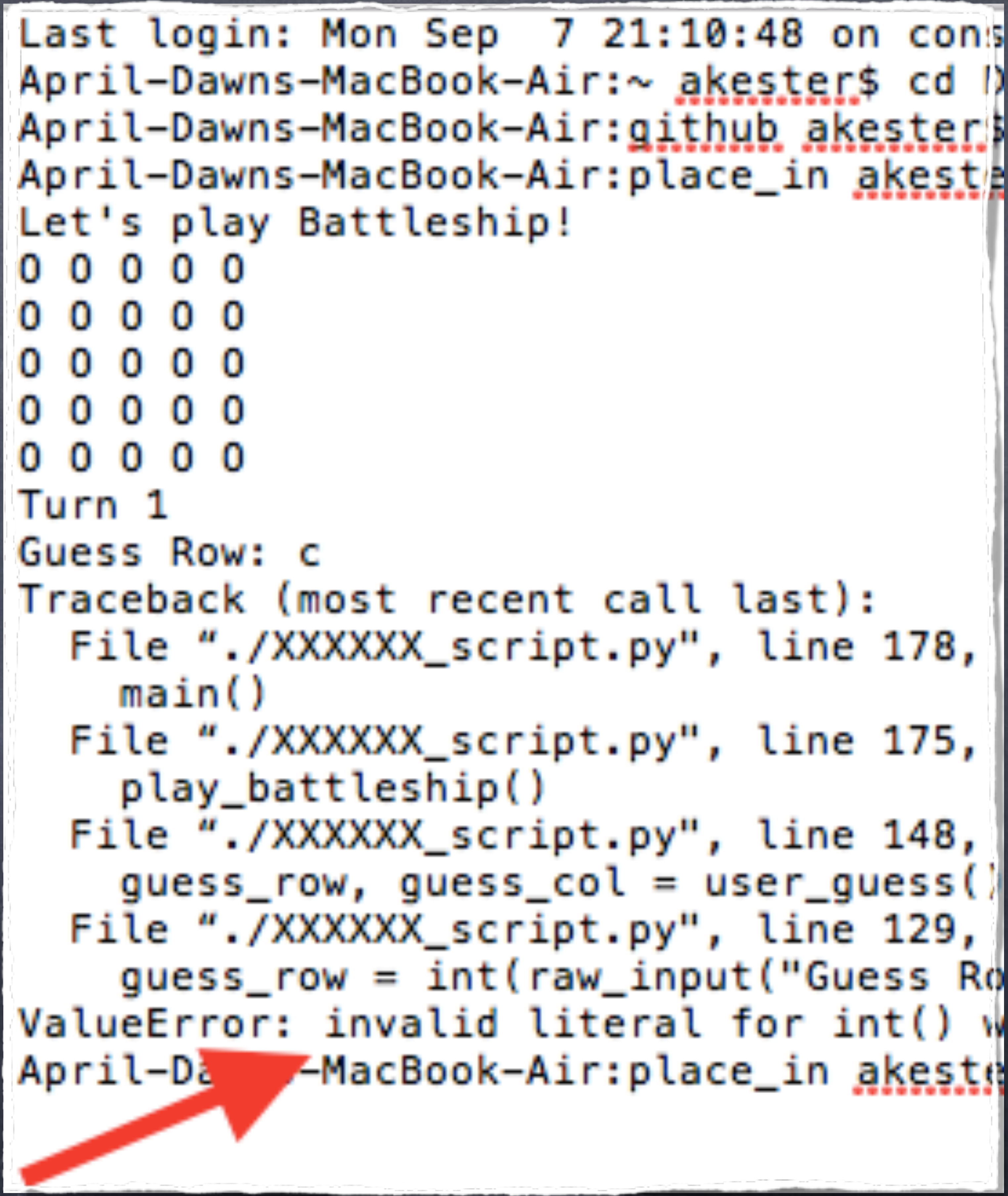
This is BAD

Dont let your program do this!



```
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

Current turn: 1
Guess Col:17
Guess Row:17

You enter the debug mode,
below is the actual battleship deployment:

Coordinate of Ship is: (2, 2)
Coordinate of Ship is: (3, 4)
0 0 0 0 0
0 0 0 0 0
0 0 X 0 0
0 0 X 0 0
0 0 X X X

Current turn: 1
Guess Col:13
Guess Row:13
Traceback (most recent call last):
  File "XXXXXXX_script.py", line 95, in <mod
    elif answer[grow][gcol] == "X":
IndexError: list index out of range
April-Dawns-MacBook-Air:place_in akester$
```

Or this...



```
Last login: Mon Sep  7 21:10:48 on con
April-Dawns-MacBook-Air:~ akester$ cd 
April-Dawns-MacBook-Air:github akester
April-Dawns-MacBook-Air:place_in akest
Let's play Battleship!
O O O O O
O O O O O
O O O O O
O O O O O
O O O O O
Turn 1
Guess Row: c
Traceback (most recent call last):
  File "./XXXXXX_script.py", line 178,
    main()
  File "./XXXXXX_script.py", line 175,
    play_battleship()
  File "./XXXXXX_script.py", line 148,
    guess_row, guess_col = user_guess(
  File "./XXXXXX_script.py", line 129,
    guess_row = int(raw_input("Guess Ro
ValueError: invalid literal for int() w
April-Da   -MacBook-Air:place_in akest
```

So what are "edge cases"?

- Edge case: occurs at an extreme (min/max) operating parameter
- Corner case: occurs outside of normal operating parameters
- Boundary case: occurs when one of the inputs is at or beyond min/max limits

Some low level edge cases

- Off by one errors
- Handling null/empty
- Infinite loop

Tips

- Go through a checklist of common mistakes
- Have an adversarial attitude toward the algorithm

Tips from MIMS '16

- "Try to break your shit before you turn it in."
- "Don't leave it until the last night."
- "It is not magic"
- "Close your laptops and pay attention in lecture"
- Take an early stab by yourself so you can go to office hours with intelligent questions. Hang out in the colab, work with people that have the same problem as you.
- "Stackoverflow can help sometimes"
- "Don't be too overwhelmed"

Tips from MIMS '16

- Start with the BIG problem, write high level logic first as a series of functions that don't do anything yet. Replace with the internal logic (could be another set of functions). Rinse and repeat.
- "You can test for a standard list of things that saves you some points on every assignment"
- Many web resources and library error messages are written for people with technical experience, it is not your fault if you're confused.

Homework Assignment #1

Grading

- Part I
- Meeting all 4 rules with corresponding password strength (1 mark)
- Printing what conditions are met and not met (2 marks)
- Responding to finish (1 mark)
- Extra credit (1 mark)

- Part II
- Checking the list (2 marks)
- Using Binary Search to check the list (2 marks)
- Counting the number of comparisons (1 mark)
- Counting correctly - as per Binary Search (1 mark)
- Extra credit (1 mark)

Getting started!

Part 1

- Reading the password from the user - read Python Input/Output
- Checking the password strength - use the in-built python methods!
- Regular expressions may help.

Part 2

- File Input
- Implementing Binary Search as a separate function will help

Some more recommendations:

1. If you didnt do the bootcamp and new to python, do an online tutorial ASAP.
2. Object Oriented programming.
3. Expected coming into the class.

Resources

- How to Think Like a Computer Scientist
- Python Documentation
- stackoverflow
- stuck.md
- Choosing the Best Python IDE
- The Art of Debugging with GDB, DDD, and Eclipse
- Sample list of edge case
- Data Abstraction and Problem Solving